REMARKS

This paper is responsive to the Non-Final Office Action dated June 2, 2004. Claims 1 – 29 were examined. Claims 1 – 29 were rejected. Applicant has amended claims 1, 3, 16, 21, 25, and 29, and added claim 30. The amendments to claims 3 and 21 are non-narrowing. Applicant respectfully traverses all of the rejections.

Rejections under 35 U.S.C. §102

Claims 1, 16, 25, and 29 stand rejected under 35 U.S.C. §102(a) as anticipated by "DCAS-Based Concurrent Deques" by Ole Agesen et al. (hereinafter *Agesen*). Claims 1 – 8, 10 – 22, and 25 – 29 stand rejected under 35 U.S.C. §102(b) as anticipated by "A Lock-Free Multiprocessor OS Kernel" by Henry Massalin and Calton Pu (hereinafter *Massalin*). Applicant respectfully traverses all of these rejections.

Single Target Synchronization Primitive

Dual target compare and swap operations, such as `DCAS` and `CAS2`, are powerful yet complex to support. Supporting simultaneous access of multiple memory locations presents certain challenges. In fact, many conventional multiprocessor architectures do not support `DCAS` or `CAS2` operations. In contrast, single target primitives, such as the compare-and-swap (CAS) instruction and load lock/store-and-compare (LL/SC) instruction pair, are widely supported. Applicant has amended the independent claims 1, 16, 25, and 29 to emphasize that the synchronization primitive(s) as recited in the claims atomically examine and update a single target. The updating is conditional updating based on the examination. Thus, Applicant's disclosure provides a highly usable implementation of a concurrent shared object with non-blocking, linearizable operations.

Neither *Massalin* nor *Agesen* disclose or suggest implementing a concurrent shared object using a single target synchronization primitive as recited in Applicant's claims. *Massalin* discloses a `Delete` algorithm that employs `CAS2` (Figure 2, page 5, section 3.3), which is a dual target instruction. *Massalin* employs a `CAS` instruction for `Insert` and for `ReleaseNode` to decrement a reference counter (Figure 2, page 5; Figure 3), but not for logical deletion. *Massalin* does not disclose or suggest using a single target synchronization primitive "to encode a marked

node indication signifying logical deletion of a corresponding one of the values from the group" as recited in claims 1, 25, and 29, or "a first functional sequence performing a logical deletion of the value using a synchronization primitive to mark a corresponding one of the nodes" as recited in claim 16. *Agesen* discloses a concurrent dequeue implemented with a DCAS operation, which is a dual target operation. *Agesen* does not disclose or suggest "linearizable operations...wherein concurrent execution of the linearizable operations is mediated using a first synchronization primitive...wherein the first synchronization primitive atomically examines and updates a single target, the updating being conditional on the examination" as recited in claim 1 and similarly in claims 25 and 29, or "the first functional sequence performing a logical deletion of the value using a synchronization primitive... wherein the synchronization primitive atomically examines and updates a single target, the updating being conditional on the examination" as recited in claim 16.

Additional failure of *Massalin* to anticipate claims

With regard to claims 6, 7, and 29, *Massalin* does not disclose or suggest "traversing the encoded group without use of an atomic operation" as recited in claim 29, "traversal of the concurrent shared object is without atomic update of a garbage collection coordination store" as recited in claim 6, or "mere traversal of the concurrent shared object is without atomic update of the concurrent share object" as recited in claim 7. The traversal code in *Massalin* includes a CAS2 instruction, which is an atomic instruction (Figure 3). The Office Action refers to the 4th full paragraph of page 3, which discloses separation of a run queue traversal and the queue element update. On page 6, in section 3.3., *Massalin* specifically states that a 2-word Compare-and-Swap is utilized to set a mark in the run-queues implementation upon entering a node, which occurs during traversal of a linked-list of nodes.

With regard to claims 1 and 25, *Massalin* does not disclose or suggest "linearizable operations to define semantics of at least insert and remove operations" as recited in claim 1 or "wherein instances of the functional sequences are linearizable" as recited in claim 25. Although linearizable and atomic are not the same, despite the assertions in the Office Action, *Massalin* does not disclose linearizable operations or atomic operations for insertion and removal of values from a concurrent shared object. *Massalin* states that "[t]he main difficulty with linked list

traversal is that nodes can disappear while visiting them" (page 6, section 3.3), which fails linearizability and atomicity. *Massalin* proposes an alternative, which is utilizing a binary marker to restrict access. This binary marker technique is essentially a locking technique, thus blocking (page 6, section 3.3). However, claim 1 recites a "non-blocking concurrent shared object" and claim 25 recites "at least two functional sequences providing non-blocking access to a concurrent shared object."

With regard to claim 3, *Massalin* discloses a CAS2 instruction, which is not a single target synchronization primitive.

With regard to claim 4, the section relied upon by the Office Action (4th full paragraph on page 3) simply states that the run-queue traversal is separate from the queue element update. There is no disclosure in *Massalin* regarding reclamation of storage associated with the excised node being independent of linearizable operations as recited in claim 4.

With regard to claim 14, the Office Action assumes without any support that a binary marker, if characterizable as a distinguishing bit value, is in an unused portion of a next node pointer of the logically deleted node. Nothing in the section cited by the Office Action indicates where the binary marker is located. The relied upon section of *Massalin* only states "[w]e set the mark at the same time we enter the node using a two-word Compare-and-Swap."

With regard to claim 15, the Office Action states that a "two-word Compare-and-Swap can guarantee safety by simultaneously checking the previous node's pointer." Neither the statement in the Office Action nor *Massalin* discloses or suggests utilizing any levels of indirection and especially does not disclose or suggest "the marked node indication includes a distinguishing additional level of indirection between the next node point of the logically deleted node and a respective other one of the nodes" as recited in claim 15.

None of the art of record discloses or suggests any of Applicant's claims. Accordingly, the independent claims and dependents therefrom are allowable for at least the reasons given above.

Rejections under 35 U.S.C. §103(a)

Claims 19, 23, and 24 stand rejected under 35 U.S.C. §103(a) as unpatentable over Massalin in view of U.S. Patent No. 6,581,063 naming Richard Karl Kirkman as an inventor (hereinafter *Kirkman*). Applicant respectfully traverses all of these rejections. Claims 9, 23 and 24 are at least allowable because they depend from corresponding ones of the above allowable independent claims.

None of the art of record discloses or suggests any of Applicant's claims. Accordingly, the independent claims and dependents therefrom are allowable for at least the reasons given above.

Conclusion

In summary, claims 1 – 30 are in the case. All claims are believed to be allowable over the art of record, and a Notice of Allowance to that effect is respectfully solicited. Nonetheless, if any issues remain that could be more efficiently handled by telephone, the Examiner is requested to call the undersigned at the number listed below

CERTIFICATE OF MAILING OR TRANSMISSION

I hereby certify that, on the date shown below, this correspondence is being

☐ deposited with the US Postal Service with sufficient postage as first class mail, in an envelope addressed to Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.

☒ facsimile transmitted to the US Patent and Trademark Office.

Steven R. Gilliam — 2-Sep-2004 (Date)

EXPRESS MAIL LABEL: _____

Respectfully submitted,

Steven R. Gilliam, Reg. No. 51,734
Attorney for Applicant(s)
(512) 338-6320
(512) 338-6301 (fax)